Computer and Information Science
www.ccsenet.org/journal.html
Vol. 2, No. 1
February 2009

Pixel Based Temporal Analysis Using Chromatic Property for Removing Rain from Videos

Peng Liu
College of Computer Science and Technology, Harbin Institute of Technology
PO box 352, Harbin Institute of Technology, China
Tel: 86-451-8641-3631 E-mail: pengliu@hit.edu.cn

Jing Xu (Corresponding author)
College of Computer Science and Technology, Harbin Institute of Technology
PO box 352, Harbin Institute of Technology, China
Tel: 86-451-8641-3631 E-mail: xujing.hit@gmail.com

Jiafeng Liu
College of Computer Science and Technology, Harbin Institute of Technology
PO box 352, Harbin Institute of Technology, China
Tel: 86-451-8641-3631 E-mail: jefferyliu@hit.edu.cn

Xianglong Tang
College of Computer Science and Technology, Harbin Institute of Technology
PO box 352, Harbin Institute of Technology, China
Tel: 86-451-8641-3631 E-mail: txl60@public.hr.hl.cn

The research is financed by National Natural Science Foundation of China.No.60702032 (Sponsoring information), and by Natural Scientific Research Innovation Foundation in Harbin Institute of Technology. No.HIT.NSRIF.2008.63 (Sponsoring information)

Abstract

The raindrops degrade the performance of outdoor vision system, and it brings difficulties for objects detection and analysis in image sequence. In this paper, we propose an algorithm detect moving objects using chromatic based properties in rain-affected videos captured by outdoor vision systems. Thus the raindrop removal algorithm includes two parts that is removal raindrops in background and removal raindrops in moving objects. Since the degradation made by raindrops is complex and appears as various changes. The raindrops detection function considering the chromatic properties of image sequence is induced, which does not need the velocity and time information of raindrops. Therefore, it is suitable for all the blur effects caused by raindrops. The removal raindrops are able to distinguish accurately the raindrops-affected pixels from the immovable or movable objects. Although the objects are moving in the rain, the algorithm is also effectual. The experiment results show that the proposal algorithm is able to remove the raindrops and improve the quality of image sequence remarkable.

Keywords: Imaging model, Complex scenes, Video processing, Outdoor vision and weather

1. Introduction

The rain-affected image sequences annoy human viewers and degrade image quality. The degraded images also decrease performance of computer vision algorithm in areas such as object detection, tracking, segmentation, video surveillance.

Now many algorithms are proposed to remove obvious rain streaks. But there are still two difficult points needing further research.1. To remove light rain. Rain streaks are too small to be detected in single frame when the background is complex, but viewer can feel rain streaks in videos. 2. To get accurate intensity value. Due to the imaging precision and video compression algorithms, it's difficult to get the true intensity value of a rain-affected pixel. The existing algorithms just have good effectiveness for detecting and removing obvious and clear rain streaks. So to resolve the two difficult points described above, in this paper, we propose a general pixel based algorithm using chromatic property for removing rain from an ordinary video.

1.1 Previous work

The methods of removing rain streaks can be classified into two types: pixel-based method and frequency-based method. The first pixels-based method used a temporal median filter in (Starik, 2003). This works during moderate rain in which the scenario is not seriously corrupted. But moving objects will cause blur effects.

For removing rain streaks during video acquisition, Garg and Nayar (2007, pp.3-27) proposed modifying the camera parameters by increasing exposure time or reducing the depth of field. But for some types of cameras their parameters are not adjustable. For removing rain streaks in video, Garg and Nayar (2004) supposed that there are few raindrops will cover three consecutive frames. So if a pixel is covered by one raindrop, the intensity change of this pixel between this frame and its previous frames is equal to the intensity change between this frame and its latter frame. Then they used a linearity constraint to reject improperly detected pixels. Finally they calculated the binary rain field to segment rain area. This method has two limits: 1. In heavy rain, raindrops could frequently affect the same position in three consecutive frames. 2. Due to noise or video compress algorithm, the linearity constraint is not always valid.

Zhang at al (2006, pp.461-464) proposed another pixel-based algorithm using the chromatic constraint. They found If a pixel is covered by raindrops, the varieties of the intensities of R , G , and B , are approximately the same. This property is used to segment moving objects and rain area. But in some videos, the intensity changes of R , G , and B in area where comprise object motion are far less than in rain-affected area. It is difficult to find an appropriate threshold that is suitable for both stationary and dynamic objects.

Barnum at al (2007) used frequency information to detect and remove rain streaks. They used a blurred Gaussian to approximate the blurred effect caused by a moving raindrop. It works when the streaks are clear. But for light rain or heavy rain, the model of a blurred Gaussian is not always effective.

1.2 Our work

This paper focuses on rain removal in video. If a video is captured by a moving camera, the frames can be stabilized in advance. So to make the problem simpler, we mainly focus on scenario comprising a stationary background and some moving objects captured by a stationary camera.

First, we make a further study of the raindrop's model, and give a general detecting method using chromatic property which is able to avoid noise effect. Then we estimate the distribution of the detected variable to reject improperly detected pixels. In the removal step, we use the imaging model to restore the background information. In this paper, we don't use the information of raindrops' velocities or shapes. Therefore our method is suitable for various types of raindrops. The result shows that our method is effective and has a better performance.

2. Chromatic properties of rain streaks

Garg and Nayar (2007, pp.3-27) got a conclusion that when a raindrop is passing through a pixel, the intensity of its image is brighter than the background. This imaging process is illustrated by the formula:

$$I_r(x,y) = \int_0^{\tau} E_r(x,y)dt + \int_{\tau}^{T} E_b(x,y)dt \tag{1}$$

Where I_r is the intensity of this pixel affected by the raindrop. τ is the time during which the raindrop projects onto the pixel. E_r is the irradiance caused by the raindrop during the time τ , E_b is the average irradiance of the background. T is the exposure time of the camera.

If the background is stationary, or the motion of it is slow, we can use the average irradiance value E_b to calculate the background intensity of the pixel I_b over the time duration T .

$$I_b(x,y) = T \cdot E_b(x,y) \tag{2}$$

Also we use the time-averaged irradiance $\bar{E}_r(x,y) = \frac{1}{\tau}\int_0^{\tau} E_r(x,y)dt$ to compute the intensity of raindrops.

$$I_r(x,y)=\tau\bar{E}_r(x,y)+\frac{T-\tau}{T}I_b(x,y) \tag{3}$$

Use $\Delta I(x,y)=I_r(x,y)-I_b(x,y)$ to denote the change in intensity at a pixel due to a raindrop, we obtain

$$\Delta I(x,y)=I_r(x,y)-I_b(x,y)=\tau\bar{E}_r(x,y)-\frac{\tau}{T}I_b(x,y) \tag{4}$$

Let $I_E(x,y)=T\bar{E}_r(x,y)$, $\alpha=\frac{\tau}{T}$, rewrite Eq.(3) and Eq.(4)

$$I_r(x,y)=\alpha I_E(x,y)+(1-\alpha)I_b(x,y) \tag{5}$$

$$\Delta I(x,y)=\alpha I_E(x,y)-\alpha I_b(x,y) \tag{6}$$

Here $I_E(x,y)$ means the equivalent ideal intensity caused by raindrops during the exposure time T. $\alpha=\tau/T$ means the ratio of rain-affected time to the exposure time.

From Eq.(6) we know that the intensity change of pixels along a rain streak is proportional to its background values. But usually the intensity values of pixels along a rain streak are similar.

$$\alpha=\frac{\Delta I_1(x,y)-\Delta I_2(x,y)}{I_{b2}(x,y)-I_{b1}(x,y)} \tag{7}$$

proximate to the intensity of a little white sphere. Therefore the equivalent ideal intensity $I_E(x,y)$ and the ratio α are uniform respectively in R, G, B channels. That is, although a falling raindrop appears as complicate shape and intensity change along the falling direction, the values of $I_E(x,y)$ and α at each pixel are equal respectively in three channels. But the Eq.(6) shows the relation of rain-affected pixel and its background. When rain is heavy, raindrops usually cover the same position in consecutive frames**.** It's hard to get the accurate value of background. In the next part, we will derive the formula of the relation between rain-affected pixels in consecutive frames.

2.1 Relation of Background pixel and rain pixel

Consider pixels at the same position in two consecutive frames, one is background pixel, and the other is rain-affected pixel. The brighter one is rain-affected pixel. $\Delta\vec{I}(x,y)$ denotes the vector of intensity change between frames in R, G, B channels. $\vec{I}_b(x,y)$ is the vector of background intensity in three channels. We rewrite the Eq.(6)

$$\Delta\vec{I}(x,y)=\alpha\vec{I}_E(x,y)-\alpha\vec{I}_b(x,y) \tag{8}$$

Where $\vec{I}_E(x,y)=[I_{ER}(x,y),I_{EG}(x,y),I_{EB}(x,y)]^T$.

From the discussion above, we get $I_{ER}(x,y)=I_{EG}(x,y)=I_{EB}(x,y)$. We still use $I_E(x,y)$ to denote this value.

2.2 Relation of Two Rain Affected Pixels

When rain is heavy, or the frames extracted from videos are not consecutive frames. Sometimes the pixels at position (x,y) in both frames are covered by raindrops. We use $\vec{I}_r(x,y)=[I_{rR}(x,y),I_{rG}(x,y),I_{rB}(x,y)]^T$ denotes the intensity vector of the brighter one, use $\vec{I}'_r(x,y)=[I'_{rR}(x,y),I'_{rG}(x,y),I'_{rB}(x,y)]^T$ denotes the other one. $\vec{I}_b(x,y)=[I_{bR}(x,y),I_{bG}(x,y),I_{bB}(x,y)]^T$ is its background intensity vector. $\vec{I}_E(x,y)=[I_{ER}(x,y),I_{EG}(x,y),I_{EB}(x,y)]^T$ and α is the parameters according to the brighter pixel. $\vec{I}'_E(x,y)=[I'_{ER}(x,y),I'_{EG}(x,y),I'_{EB}(x,y)]^T$ and α' is the parameters of the other one. We get

$$\vec{I}_r(x,y)=\alpha\vec{I}_E(x,y)-(1-\alpha)\vec{I}_b(x,y) \tag{9}$$

$$\vec{I}'_r(x,y)=\alpha'\vec{I}'_E(x,y)-(1-\alpha')\vec{I}_b(x,y) \tag{10}$$

From Eq.(10), we get $\vec{I}_b(x,y)=-\frac{\vec{I}'_r(x,y)-\alpha'\vec{I}'_E(x,y)}{1-\alpha'}$, substitute it into Eq.(9), we get

$$\vec{I}_r(x,y)-\vec{I}'_r(x,y)=\alpha\vec{I}_E(x,y)-\frac{1-\alpha}{1-\alpha'}\alpha'\vec{I}'_E(x,y)-\frac{\alpha-\alpha'}{1-\alpha'}\vec{I}'_r(x,y) \tag{11}$$

Comparing Eq.(11) with Eq.(8), it shows that Eq.(8) is the special form of Eq.(11) when $\vec{I}'_r = \vec{I}_b$. In this condition the pixel with less intensity value is background pixel. And the rain-affected time is zero, so $\alpha' = 0$.

Since the terms of $\alpha\vec{I}_E(x,y) - \frac{1-\alpha}{1-\alpha'}\alpha'\vec{I}'_E(x,y)$, and $\frac{\alpha-\alpha'}{1-\alpha'}$ in Eq.(11) are equal respectively in three channels, let $\vec{A} = \alpha\vec{I}_E(x,y) - \frac{1-\alpha}{1-\alpha'}\alpha'\vec{I}'_E(x,y)$, $K = \frac{\alpha-\alpha'}{1-\alpha'}$, we can rewrite Eq.(11)

$$\Delta\vec{I}(x,y) = \vec{A} - K\cdot\vec{I}'_r(x,y) \tag{12}$$

Eq.(12) is the pixel-based imaging formula using chromatic properties.

3. Existence of solution and noise effect

Eq.(12) is the imaging model of rain-affected pixels which is used as a detecting function. But when $\vec{I}'_r(x,y)$ is affected by noise, it is difficult to get an accurate value so that K is inaccurate. Let $\vec{\xi} = [\xi_{R,}\xi_{G,}\xi_{B,}]^T$ denote the noise in three channels, from Eq.(12), we get

$$\Delta\vec{I}(x,y) = \vec{A} - K\cdot\vec{I}'_b(x,y) + \vec{\xi} \tag{13}$$

Using two channel to calculate the value of K, for example using R, G channels:

$$K = -\frac{\Delta I_R(x,y) - \Delta I_G(x,y)}{I'_{bR} - I'_{bG}} + \frac{\xi_R - \xi_G}{I'_{bR} - I'_{bG}} \tag{14}$$

Calculation of K is of two types.

(1) $I'_{bR} - I'_{bG}$ does not approach to zero

When $I'_{bR} - I'_{bG}$ does not approach to zero, suppose $I'_{bR} - I'_{bG} \geq C$, and $C > 0$. The term of noises is negligible. Using Eq.(12), we will get three K values respectively in R, G channels. So according to the imaging model of pixels covered by raindrops, if the three K values are similar, this pixel is a rain-affected pixel. Otherwise it belongs to a moving object.

(2) $I'_{bR} - I'_{bG}$ approaches to zero

When $I'_{bR} - I'_{bG}$ approaches to zero, the term of noises is not negligible. Although a pixel is rain-affected pixel, the three K values calculating by three channels may be not equal to each other. In this condition, $(I'_{bR} - I'_{bG}) \to 0$, $K = (\alpha-\alpha')/(1-\alpha')$, where $\alpha = \tau/T$, due to the fast movement of raindrops, $0 < \tau \ll T$, therefore $|K| < 1$, we get $K(I'_{bR} - I'_{bG}) \to 0$. From Eq.(13), when a pixel is affected by a raindrop, $(\Delta I_R(x,y) - \Delta I_G(x,y)) \to (\xi_R - \xi_G)$. Since the noise is caused by illumination or imaging process, it's far less than the intensity change caused by a mobbing object. We get $\xi_R - \xi_G \to 0$. So if we use k and k' to denote different channels, when $\Delta I_k(x,y) - \Delta I_{k'}(x,y)$ approaches to zero, the pixel at (x,y) is affected by raindrops, otherwise it's of a moving object.

4. Segmentation of moving object

From the discussion above we know if a pixel is covered by a raindrop, either $\Delta I_R(x,y) - \Delta I_G(x,y) \to 0$, or $I'_{bR}(x,y) - I'_{bG}(x,y)$ does not approach to zero. So the equation $K = -(\Delta I_R(x,y) - \Delta I_G(x,y))/(I'_{bR}(x,y) - I'_{bG}(x,y))$ has a solution. We use a general formula to calculate the variable k.

$$K = J(\Delta I_R(x,y) - \Delta I_G(x,y)) - (1-J)\frac{\Delta I_R(x,y) - \Delta I_G(x,y)}{I'_{bR}(x,y) - I'_{bG}(x,y)} \tag{15}$$

Where $\begin{cases} J = 1, \text{ when } I'_{bR}(x,y) - I'_{bG}(x,y) \leq C \\ J = 0, \text{ when } I'_{bR}(x,y) - I'_{bG}(x,y) > C \end{cases}$, C is a threshold to judge if $I'_{bR}(x,y) - I'_{bG}(x,y)$ approaches to zero. For three channels we will get three k values by using Eq.(15). That is

$$\begin{cases} K_1 = J_1(\Delta I_R(x,y) - \Delta I_G(x,y)) - (1-J_1)\dfrac{\Delta I_R(x,y) - \Delta I_G(x,y)}{I'_{bR}(x,y) - I'_{bG}(x,y)} \\ K_2 = J_2(\Delta I_R(x,y) - \Delta I_B(x,y)) - (1-J_2)\dfrac{\Delta I_R(x,y) - \Delta I_B(x,y)}{I'_{bR}(x,y) - I'_{bB}(x,y)} \\ K_3 = J_3(\Delta I_B(x,y) - \Delta I_G(x,y)) - (1-J_3)\dfrac{\Delta I_B(x,y) - \Delta I_G(x,y)}{I'_{bB}(x,y) - I'_{bG}(x,y)} \end{cases} \tag{16}$$

The final value of k is $\max(k_1, k_2, k_3)$ .To determine whether a pixel is covered by a raindrop by values of k is very tedious. Here we use another method to segment moving object from rain field.

From Eq.(11) and Eq.(12), we know $K = (\alpha - \alpha')/(1-\alpha')$. Consider $K-\alpha$, because $K - \alpha = \alpha' \cdot (\alpha - 1)/(1-\alpha')$, and $0 < \alpha \ll 1$ we get $K - \alpha < 0$, that is $K \le \alpha$. So

$$K^n \cdot \Delta I(x,y) \le \alpha^n \cdot \Delta I(x,y) < \alpha^{n+1} \cdot I_E(x,y) \tag{17}$$

Where $\alpha = \tau / T$, and $\tau \ll T$. So when the pixel at (x,y) is covered by a raindrop, we have $\lim_{n\to\infty} \alpha^{n+1} \cdot I_E(x,y) \to 0$. However when the pixel belongs to a moving object, the term of $K^n \cdot \Delta I(x,y)$ does not converge to zero. The detecting function is

$$F(x,y) = K^n \cdot \Delta I(x,y) \tag{18}$$

If $F(x,y)$ dose not converge to zero, the pixel is of a moving object. We obtain the edges of moving object by using this detecting function.

5. Detection and removal of rain

As mentioned in section 1.2, we suppose those frames abstracted from video are aligned in advance. Using the detecting function, a frame is divided into two parts: static background and foreground. So the removal algorithm is twofold:

(a) To remove raindrops in background.

(b) To remove raindrops in foreground.

Many methods are suitable for removing raindrops in static background. We use Kalman Filter to suppress intensity increase caused by raindrops. To remove raindrops in foreground, we align the foregrounds respectively, and then use the method of removing raindrops in background like the first step.

6. Results and Analyses

Our experiments use a threshold of 3 gray levels to detect the intensity change of pixels. Use a threshold of 5 gray to judge if the subtraction between channels such as $I'_{bR}(x,y) - I'_{bG}(x,y)$ approaches to zero. In calculating of detecting function, the order of K is 1, the threshold of the detecting function is 5.

Fig.2 (a) is an image of static scene from the video captured by Zhang and Li (2006, pp.461-464). (b) is removal result using Garg and Nayar's method (2007, pp.3-27). The result shows that the method of Garg and Nayar is not effective when rain is heavy because their removal algorithm only calculates two consecutive frames. Image using method of Zhang and Li (2006, pp.461-464) has better quality. But their method uses K-means clustering to calculate the background color which is effective in static scene. But this method will damage the image quality in some areas of dynamic scene by improperly regarded some moving object pixels as rain-affected pixels.

Fig. 3 shows results in dynamic scenes. Fig. 3 (b) and (c) are result of Zhang's method. The improperly detected pixels obviously damage the visual quality of derained image. (d) is our result. (e) and (f) are local areas of (d). Compared with (c), our results accurately detected rain-affected area. (g) is another frame in this video. (j) is derained frame. (h) and (i) are local areas of (g). (k) and (l) are local areas of (j). Our results show a better performance in this conditon.

7. CONCLUSIONS

By further studying the model of raindrops, we obtain a detecting function using chromatic property which is suitable for a general rain condition. It can segment rain-affected pixels from moving objects between two frames. Then removing approach is classified into two steps: to remove raindrops in background and to remove raindrops covered moving objects. The removal method makes the pixels at the same position of backgrounds in each frame similar and shows a better visual quality. Another advantage is that our method does not use any information about the shape, the velocity of raindrops, neither uses the value of camera's exposure time. Therefore it is effective in various conditions, such as heavy rain, light rain, rain in focus, rain out of focus and so on. When the objects are moving in the rain, this is also effective. Moreover the method of Garg (2007, pp.3-27) and Zhang (2006, pp.461-464) need many consecutive frames to calculate the information of rain-affected area. However our algorithm is able to segment rain field from moving object in two arbitrary frames. The results show that our method has better visual quality.

References

Barnum, P, Kanade, T., and Narasimhan S., (2007). “Spatio-Temporal Frequency Analysis for Removing Rain and Snow from Videos, ” *PACV workshop at ICCV*.

Braillon, C., Pradalier, C., Crowley, J.L., Laugier, C., (2006). “Real-time moving obstacle detection using optical flow models.”*Proc. 2006 IEEE Intelligent Vehicles Symposium*, IV 2006, pp. 466-471.

Cucchiara, R., Grana C., Piccardi M., and Prati A., (2003). “Detecting moving objects, ghosts, and shadows in video streams,” *IEEE Trans on Pattern Anal. and Machine Intel*., vol. 25, no. 10, pp. 1337-1442,2003.

C. Wren, A. Azarhayejani, Darrell, T., and Pentland, A.P., (1997). “Pfinder: real-time tracking of the human body,” *IEEE Trans. on Pattern Anal. and Machine Intell*., vol. 19, no. 7, pp. 780-785, 1997.

C. Ulbrich, W.. (1983). “Natural variations in the analytical form of the raindrop size distribution,” *J. of Applied Meteorology,* 22(10):1764–75, 1983.

Garg, K., and Nayar, S. K., (2004). “Detection and removal of rain from videos,”*CVPR*, 2004.

Garg, K., and Nayar, S. K., (2006). “Photorealistic rendering of rain streaks,” *SIGGRAPH*, 2006.

Garg, K., and Nayar, S. K., (2007). “Vision and Rain,”*IJCV*, pp.3-27,2007.

Han, B., Comaniciu, D., and Davis, L.S.,(2004). “Sequential kernel density approximation through mode propagation: applications to background modeling,” *Proc. Asian Conf. on Computer Vision*, Jan. 2004.

Liu, Polley R., Meng, Max Q.-H., Liu, Peter X., Tong, Fanny F. L., Wang, Xiaona, (2006). “Optical flow and active contour for moving object segmentation and detection in monocular robot,” *Proc. IEEE International Conference on Robotics and Automation*, ICRA 2006, pp. 4075-4080.

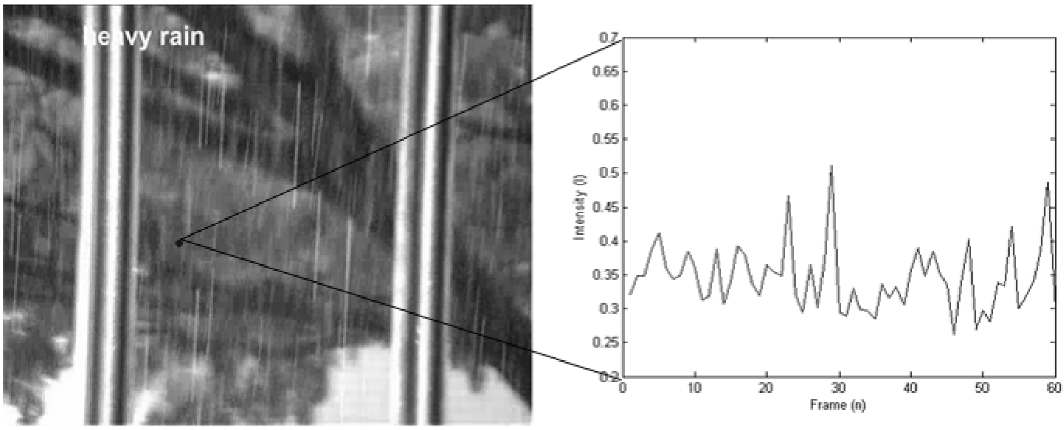
Narasimhan, S.G. and Nayar, S.K. (2002). “Vision and the Atmosphere,” IJCV, 2002, 48(3):233–254.

Starik, S., and Werman, M., (2003). “Simulation of rain in videos”. In Int’l. *Workshop on Texture Analysis and Synthesis*, 2003.

Seki, M., Wada, T., Fujiwara, H., and Sumi, K., (2003) “Background subtraction based on cooccurrence of image variations,” *Proc. CVPR 2003*, Vol. 2, pp. 65-72,2003.

Wayne, P. Power, and Schoonees, J. A., “Understanding background mixture models for foreground segmentation”, *Proc. of JVCNZ 2002*, pp. 267-271, Nov. 2002.

Zhang, X., Li, H., Qi, Y., Kheng, W., and Ng., T. K., (2006). “Rain removal in video by combining temporal and chromatic properties.” *ICME*, pp.461-464, 2006.

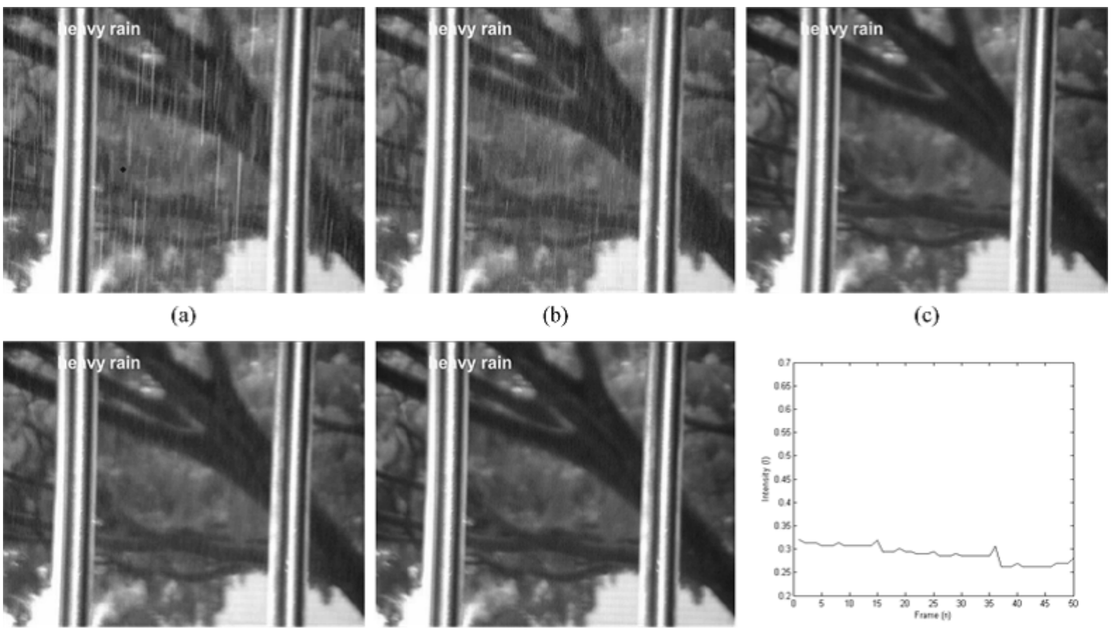


(a) (b)

Figure 1. Rain affected frame and the intensities change at one fixed position in this video.

(a) is a rain-affected frame with stationary background in heavy rain.

(b) shows the intensities change of the position in (a)



(a) (b) (c)

(d) (e) (f)

Figure 2. One frame in static scene from the video captured by Zhang and Li.

(a) is the original frame. (b) is the result using Garg's method. The removal algorithm uses three frames. (c) is the result using Zhang's method. The removal algorithm uses 200 frames. (d) is the result using our method. The removal algorithm uses 10 frames. (e) is the result of our method using 50 frames. (f) is the intensities change of the de-rained frames on the position in (a), the removal algorithm uses 10 frames

Figure 3. Some experiment results in dynamic scenes.

The video is a chip from the movie "Magnolia". It has been used by Garg and Zhang. (a) is the original frame. (b) is the result of Zhang's method. (c) is a local area of (b). (d) is our result calculating 10 frames. (e) and (f) are local areas of (d). (g) is another frame of this video. (h) and (i) are local areas of (g). (j) is our result calculating 10 frames. (l) and (k) are local areas of (j)